

Final Review

CPS 116
Introduction to Database Systems

2

Announcements (December 6)

- ❖ Homework #4 graded
 - Sample solution available today
- ❖ Final exam next Saturday 7-10pm
 - Open book, open notes; similar format as sample final
 - Focus on materials after the midterm
 - Sample solution available today
 - Final review session tentatively scheduled for Friday afternoon 5pm
- ❖ Handouts
 - Lecture notes and assignments: course Web site
 - Graded assignments and midterm: see me
 - Sample solutions and sample exams: available only in hardcopies; email me if you cannot find one
- ❖ Check Blackboard to make we entered your scores correctly!

3

Review: relational basics

- ❖ Relational model/algebra → physical data independence
- ❖ Entity-relationship design
- ❖ Design theory (FD's, MVD's, 3NF, BCNF, 4NF) → help eliminate redundancy
- ❖ SQL
 - NULL and three-value logic → nifty feature, big mess
 - Bag versus set semantics → careful about equivalences
 - SFW (or SPJ) queries, subqueries, grouping and aggregation
 - Modifications
 - Constraints → the more you know the better you can do
 - Triggers (ECA) → "active" data
 - Views → logical data independence
 - Indexes → reintroduce redundancy to improve query performance
 - Transactions and isolation levels

Review: XML

4

- ❖ Data model: well-formed vs. valid (DTD \approx schema)
 - ❖ Query languages
 - XPath: (branching) path expressions (with conditions)
 - XQuery: FLWOR, subqueries in return (restructuring), quantified expressions, aggregation, ordering
 - XSLT: structural recursion with templates
 - ❖ Programming: SAX (one pass) vs. DOM (in memory)
 - ❖ Relational vs. XML
 - Tables vs. hierarchies (or graphs in general)
 - Storing XML as relations
 - Schema-oblivious: node/edge based, interval based, path based, etc.
 - Schema-aware
- Joins vs. path traversals

Review: physical data organization

5

- ❖ Storage hierarchy (DC vs. Pluto) → count I/O's
 - ❖ Disk geometry: three components of access cost; random vs. sequential I/O
 - ❖ Data layout
 - Record layout (handling variable-length fields, NULL's)
 - Block layout (NSM, PAX) → inter-/intra-record locality
 - ❖ Access paths
 - Primary versus secondary indexes
 - Tree-based indexes: ISAM, B⁺-tree
- Again, reintroduce redundancy to improve performance
→ Fundamental trade-off: query versus update cost

Review: query processing, optimization

6

- ❖ Processing
 - Scan-based algorithms
 - Sort- and hash-based algorithms (and their duality)
 - Index-based algorithms
 - Pipelined execution with iterators
- ❖ Optimization (or "goodification"?)
 - Heuristics: push selections down; smaller joins first
→ Reduce the size of intermediate results
 - Cost-based
 - Query rewrite: merge blocks to get a bigger search space
 - Cost estimation: result size estimation; use statistics
 - Search algorithm: dynamic programming (+ interesting orders)

Review: transaction processing

7

- ❖ ACID properties
- ❖ Concurrency control
 - Serial and conflict-serializable schedules
 - Locking-based: 2PL, strict 2PL
- ❖ Recovery with logging
 - Steal: requires undo logging
 - No force: requires redo logging
 - WAL (log holds the truth)
 - Fuzzy checkpointing
